MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LIBRARIES

no 1958-87

1988

MIT Libraries

This material has been supplied by:

Interlibrary Loans
Massachusetts Institute of Technology
Building 14S-234
Cambridge, Massachusetts 02139-4307

Phone: (617) 253-2761
Fax: (617) 253-3109

OCLC symbol: MYG

To: 4 Wilson Library —
U. of Minnesota

PAPER
OL OF MANAGEMENT

Third Party
or Two Party Single-
-Issue Negotiations

eltner

November 1987

USETTS
ECHNOLOGY
AL DRIVE
CHUSETTS 02139

Patron: Goreigh, M.

IL #: 6145801

Date Sent: APR 1 8 1996

Date Due: 10 days for receipt

Restrictions:

✓ No Used Jiffy Bags
______No Renewals
______$15.00 Fee; Invoice to follow
______No Photocopying
______Room Use Only

Please return this form with material.

.M414
no 1958-87

WORKING PAPER
ALFRED P. SLOAN SCHOOL OF MANAGEMENT

The Computer as a Third Party

Decision Support System for Two Party Single-Issue and Two Party Multi-Issue Negotiations

Christoph Goeltner

WP 1958-87 November 1987

MASSACHUSETTS
INSTITUTE OF TECHNOLOGY
50 MEMORIAL DRIVE
CAMBRIDGE, MASSACHUSETTS 02139

ns:

____No Used Jiffy Bags

____No Renewals

____$15.00 Fee; Invoice to follow

____No Photocopying

____Room Use Only

The Computer as a Third Party

Decision Support System for Two Party Single-Issue and Two Party Multi-Issue Negotiations

Christoph Goeltner

WP 1958-87 November 1987

The author is Ph.D. candidate, Department of Ocean Engineerinng at the Massachusetts Institute of Technology.

FEB 0 8 1988

Table of Contents

Abstract page 3
Acknowledgement 4
Introduction 5
Function of the computer in negotiation 7
  Quantitative aspects 8
  Non-quantitative aspects 8

Two Party Single Issue Support System,(ONDINE I) 10
  Part I Input of negotiation paramaters 11
    Input first offer (FO) 11
    Reservation Price (RP) 12
    Estimated reservation price of the other side (RP*) 13
    Fair price (FP) 15
  Part II Negotiation 16
    Open offer 17
    Split the difference 17
    Mild contract 17
    Fair price arbitration 19

Two Party Multi Issue Support System (ONDINE II) page 22
- Part I Input of utility function 22
- Part II Negotiation 24
- Part III Improving the agreement 26
- Part IV New approaches 26

Conclusions 28

References 30

Figures 32

Abstract

Two prototype systems (ONDINE I, II) have been developed to demonstrate the potential of a computer, acting as a neutral third party to facilitate negotiations between two parties in single or multi-issue cases. In the single issue case, "fair price arbitration" and a procedure to assess the negotiators' perception about their reservation prices are implemented into a rule base in order to analyze the negotiation behaviour and enhance the effectiveness of the computer as a mediator/arbitrator. In the multi-issue case, the main role of the computer is to maximize both parties' utility, once a basic agreement has been found. New procedures to assist the multi-issue negotiation, before agreement is reached, are also discussed. The system, so far, is aimed at academic purposes, yet, it should also explore the usefulness of the computer for real world negotiations.

Acknowledgement

The project was supervised by Professor J.D. Nyhart, for whose advice and generosity I would like to express my thanks. The industrious effort of Hae Yong Yi, to implement the code and her critical remarks on the approach are most gratefully acknowledged. Project Athena provided financial and hardware support. The American Academy of Arts and Sciences funded parts of this research. The efforts of Roman Budzianowsky were essential to integrate the program.

The experience of two of the most respected experts in negotiation had a major influence on my approach. J.D. Kettelle with his idea to use the computer as a third party and Professor H. Raiffa with A. Blass who helped me with the optimization routine in ONDINE II. My thanks to all of them.

Introduction

So far, computers have been used in negotiations mostly for purposes of simulation of technical issues which were relevant to the negotiation (Nyhart, Goeltner 87). But the use of computer models is by far not limited to simulation. For example, programs have been developed to optimize utility of the negotiating parties; there are even systems which support a mediator to help him improve the parties outcome (Blass, Raiffa 86).

The idea to completely substitute a human mediator in a negotiation has to my knowledge first been proposed in the arms control field (Kettelle 84). In this particular negotiation it is highly unlikely that the super powers would trust any other nation to be a mediator. Using cryptified input data it is possible that a computer can discover possible overlaps between the parties' interests and then, according to an agreed upon algorithm suggest a range of outcomes which are acceptable to both sides, a procedure which can effectively increase the solution space. Nevertheless, such an idea has so far not been developed to an operational system, many question e.g. about structure of input and output data have not been resolved yet.

The following study and development of two computer programs is a first step to learn about the possibilities of a computer in its entirely new role of acting as an independent mediator/ arbitrator/ facilitator.

The basic idea is that the computer is able to collect private (secret) information from the negotiators without revealing it to the other side. According to an agreed upon rule base, the computer can make internally use of the information so tha the machine can give advice to the parties, where an agreement is most likely to be found. If the interests of the negotiators can be represented in a quantitative way the computer can also optimize outcomes. One main consideration of this approach is that the users are not forced into a prescribed solution, at any point they must be able to exit and reenter the system.

The first program, ONDINE I (for ONe DImensional NEgotiaiton) deals with the most simple negotiation scenario: the Two Party Single-Issue Negotiation. Optimization is in this case not possible, so the focus of the program is the Computer as a Third Party concept (CTP). It seems to be necessary to get first experience in simple scenarios, before one can address more complex negotiations.

The second program, ONDINE II, expands the scope to Two Party Multi-Issue Negotiations. The possible role that a computer can play in this scenario is much wider than in the single issue case, but the development of acceptable (to the negotiators) strategies is much more difficult and requires extensive testing.

In addition to new insights gained in negotiation research, the study serves another purpose. In the age of technology people will get more and more used to the computer's influence on every day' decisions. With the growing use of expert-systems and artificial intelligence it is important to learn about the user's reaction towards a machine which not only performs calculations, too cumbersome to be done by hand, but furthermore interacts with the user, having the potential to be a major factor for his decisions. The system described below is one example, where one could gain valuable experience. So far it has been used only in a small laboratory test (Cohen, Tong 87).

Function of the computer in negotiations

Due to its physiscal characteristics the computer was initially only used to perform numerical calculations. The emerging fields of expert systems and artificial intelligence expand the use of computers to new areas of application, although too enthusastic expectations have already been disappointed.

Negotiaiton is an area where the human, non-quantitative and non-rational elements are very pronounced and frequently dominating the dispute. Accordingly, it is difficult to convince a sceptic that the use of computers can be valuable also in this context. The way to procede is to show that the computer can do things a human being cannot do or at least in a much better (i.e. more efficient, more effective) way. Let me divide the possible tasks for a computer into two sections, one dealing with quantitative problems the other with subjective issues. Although I must point out that a clear distinction is not always possible; sometimes there is considerable overlap between these two categories. Some of the arguments in the discussion might not be strong enough, if considered separately, to justify the introduction of the computer to negotiation, but the whole range of arguements, I hope, can explain that further research in this area is important.

Quantitative aspects:

- optimization of utility of the parties, Pareto efficiency
- simulation models for the underlying issues in a negotiation:
  - *calculation of the critical parameters in different plausible scenarios (what-if); sensitivity analysis
  - *consistency check of assumptions
- provide statistics of the negotiation process, to support strategy building to improve outcome (partisan model)
- prepare negotiation in a gaming environment, supported by an expert system
- game theoretical analysis to test stability of likely outcomes and to develop negotiation strategy (partisan model)
- book keeping and organizational support for the negotiation; computer conferencing

Non-quantitative aspects:

- computer acts as third party (CTP)
- joint model building process can create mutual trust
- use of the model as a tool to improve communication
- enhance rational dimension of negotiation

The optimization of the negotiators' utility offers a good opportunity to use the computer in its traditional quantitative role. Although in some cases it is very difficult to quanitfy the objectives of the negotiators in a satisfactory way, there are many situations where the decisive issues can be described, for example in monetary terms. Then, then the computer can determine a set of optimal solutions with the potential to improve both sides' outcome (Pareto efficiency, see Raiffa 1986.).

The use of simulation models is another area that promises many opportunities for computer application in negotiation, planning and decision making. Whereas the utility concept is, at least in principle, generally applicable, simulation models have to be tailored to one specific context. Their accuracy and usefulness depends entirely on the knowledge of the modeler about the substance of the problem, which has to be simulated. Because many negotiated issues deal with problems with considerable uncertainty, a high degree of accuracy is not attainable and therefore simple models are much more likely to be acceptable.

Among the non-quantitative arguments, the use of the computer as a third party seems to me most important. If not "outperforming" a human mediator the machine could in certain standard cases substitute the expensive mediator. In some cases, as indicated in the arms control scenario, the use of the computer is possibly the only way (at least in theory) to give the negotiators a chance to use private information to expand the solution space without the intervention of a human third party, which would also not be acceptable in this case. Another non-quantitative aspect of computer models is that they can contribute to an improvement of the climate between often quite hostile opponents; this is especially the case when the parties can agree to build a computer model of the underlying issues. (See also Law of the Sea Negotiation, Sebenius 1981)

The detailed description of the two programs (see below) illustrates the choice to focus in the Two Party Single-Issue case on the implementation of CTP concepts, and in the Multi-Issue case on the optimization of utility. In this short discussion I only indicated a few reasons why computer models can be useful in negotiations but I expect that further research and practical applications will reveal more opportunities for the computer in this context. Especially promising looks the integration of simulation and optimization capabilities in an expert-system based communication network.

Two Party Single-Issue Negotiaiton Support

ONDINE I

Structure of the program:

In the first part of the session the computer asks the parties for information about the main negotiation parameters (reservation price, etc.). In the second part, the actual negotiation, the parties have several options about the involvement of the computer in the negotiation, which ranges from complete independence to arbitration. The parties are not forced to reach agreement. An operational detail of some importance is that the parties use two computer terminals and can communicate with the computer as well as with the other side. If the computers are in one room the negotiators can also communicate without using the computer. With the given hardware it is possible to support a negotiation were the parties do not see each other, they would then communicate only by sending messages to each other. It is very probable that such an environment will have a strong effect on the negotiation behaviour.

Figure I gives an overview of the parameters in the two party, single-issue negotiation as used by the program.

Part I Input of Negotiation Parameters:

Here the computer communicates separately with each side and does not reveal private information to the other party, but uses this information to give the parties advice according to a built in rule base.

The basic quantifiable parameters of a negotiation are reservation price, the offers of the parties and the agreement price. In addition to those, I introduce two new parameters which I think are useful for analysis: Fair Price (FP) and Estimated Reservation Price of the other side (RP*) which are explained below (see also figure I).

Once the negotiators are familiar with the functions of the program they start the negotiation by first giving information to the computer, the input data are the following:

Input First Offer (FO):

To ask a negotiator about his reservation price is in practice quite difficult and does not seem to be practicle. Although the negotiator is assured that this information is not revealed to the other side and that it will not influence the outcome of the negotiation he will be reluctant to reveal this information to the computer. Hence, in order to get the user accustomed to the system the computer asks first about an initial offer. These numbers are revealed to the other side simultaneously. There is no question about who goes first. In the highly unlikely case that there is an overlap (seller's price < buyer's offer) the program splits the overlap as the final outcome. In realistic cases both numbers will be far apart (buyer's offer << seller's offer) and the computer asks for further information.

Reservation Price (RP):

This price is often also circumscribed as BATNA, Best Alternative To a Negotiated Agreement. It is the maximum price the buyer would be willing to give and the minimum price the seller would accept. Although in everyday negotiations people do not use this terminology, they are usually aware of their reservation price. In this simple program the reservation price is assumed to be static, i.e. it does not change during the negotiation. Still, one should point at the existing dynamism in negotiations. Depending on incoming new information and changed perceptions of the negotiators, their reservation prices can change drastically. Nevertheless, in order to demonstrate the usefulness of the concept of a reservation price, a single issue negotiation is simple enough that it justifies the use of a static reservation price. Future systems can apply more sophisticated procedures to allow changes of the reservation price. On the other hand, complex algorithms could encourage the negotiators to develop cheating strategies by exploiting possible weaknesses in the approach, this is another advantage in using a simple procedure.

Once the numbers are put into the machine, the program compares the two reservation prices and determines whether there is an overlap, i.e. the buyer's maximum is higher than the seller`s minimum. This is the first important function the computer performes. Under the condition of roughly accurate reservation prices, the program determines whether it is worthwhile for the parties to continue the negotiation. It is probably realistic to expect that the users are not compeletely honest about their reservation prices. If the computer announces that there is no agreement possible (RP buyer < RP seller) then there are two cases which lead to this result. Either there is really no overlap and the negotiators should look for other alternatives, or they have exaggerated their reservation prices, so that there is artificially no overlap. In this situation they are asked to input their numbers again. The program limits the repetition of the input of the reservation price to two in order to avoid triggering a negotiation about the input procedure, where it is the main objective of the parties to find out about the other side's reservation price.

Estimated Reservation Price of the Other Side (RP*):

Analyses of negotiations show that the perception of the other sides situation, i.e. the reservation price, is most important. Every negotiator tries to influence the other's perception in a way that it would lead to a better agreement. (E.g. The buyer would like to appear as the "poor guy" to appeal to the seller's humanitarian responsibilities, etc.) It seems that the most important motivation to act in such a way during a negotiation is based on the principles of fairness. No matter what other rational (mostly economical) considerations might suggest otherwise, splitting one dollar into two halfs is always considered as fair. Research about the importance of "rational" fairness considerations shows that utility is always dominated by equity (Rohrbaugh 1980).

The straightforward conclusion drawn from this observation is that a deeply founded misperception of the other side's true reservation price is often the basic obstacle for an agreement. Let me illustrate this with the following example. I assume that splitting a distance into half is the basic mechanism to determine a fair outcome. Accordingly, I place a fair, i.e. acceptable outcome in the range of half the distance between my own reservation price and the one I think the other side has (RP*) (see figure II). If my perception of RP* is far off, the fair price is shifted accordingly. If the other side is not able to convince me that his reservation price is really much lower, then I am quite unlikely to come to an agreement although there is a positive bargaining range. I will not accept an outcome where the other side ends up with a much better perceived profit margin than myself. This example underlines the importance of the reservation price, and how people are motivated to manipulate the other side's perception. In order to avoid unnecessarily large misperceptions, I let the computer compare the reservation price with the estimated one. If the estimate is in the order of magnitude of the reservation price (measured in the distance between the reservation prices), it is very likely that the parties

If the computer determines that the estimate is too far off from the reservation price. The computer notifies the negotiator to indicate that he should reevaluate his assumptions.

So far I have disregarded the influence of cheating. It requires a thorough theoretical analysis and real tests to find out what strategies might be used "to beat the system". From a practicle point of view it might be justifiable to assume that in this very first stage of the use of a computer third party these considerations are of secondary importance. As long as there is no obvious bias in the system, the uncertainty about the other side's input, resulting in a type of prisoner dilemma situation, seems to be an environment safe enough to prevent obvious cheating tactics.

Fair Price (FP):

Expanding on my short discussion on fairness, I believe that every negotiator has a more or less clear idea of what a fair outcome might be, before the negotiation has even started. Again, parallel to the dynamics of the perception of the other side's reservation price, the determination of the fair price changes accordingly. In this program the computer checks whether the fair prices of both sides are within the bargaining range and gives this information to the negotiators. It is obvious that a fair, i.e. acceptable price has to lie in the bargaining range. At this point I would like to remind the reader that all information placed into the computer so far is completely secret, and the users should feel free to input their true values.

To prevent the negotiators to use this algorithm to find the other sides bargaining range, they cannot repeat inputing these numbers and iteratively reveal the other side's information. I admitt that the case where I get information that my fair price is out of the bargaining range can be a non desireable piece of information from the other side's point of view, because it gives a defined limit for the location of the other party's reservation price. Here I will leave it to further discussion, how to give information about a realistic fair price without helping one side to learn about the counterpart's private information.

At this point both parties have finished the first part of the session, they know that an agreement is possible and they got some feedback from the third party about a range of possible outcomes (concerning RP*, FP). Now the actual negotiation can start. The program gives the participants the option to chose among different negotiation styles.

Part II Negotiation

In this part of the session the system offers the user four different negotiation options. The users can switch among the different negotiation modes as long as they can both agree on which mode to chose. Two of the modes are independent of the aid of the computer: Open Offer and Splitting of the Difference. The two remaining modes allow the user to let the computer mediate or arbitrate: Mild

Open Offer:

This part of the program simply allows the negotiators to do what they would normally do in a traditional negotiation. Here they simply exchange offers to each other until they reach an agreement or they feel like using arbitration. Still, one should note that both offers are revealed to the other side at the same time (at least in this version of the program) which might in some cases be an important factor influencing the bargaining behaviour.

Split the Difference:

Strictly speaking this is not really a negotiation mode, the machine simply performes a calculation, i.e. splitting the difference between the two offers. Once, both sides' offers are close enough so that they agree to split the remaining gap, they just push a button and the result will be displayed. It should be obvious, when I speak about the difference between the offers, the buyer's offer is lower than the seller's; in the other case there is no negotiation necessary any more (see input of first offer).

Mild Contract (Kettelle 1986.2):

At first, this procedure looks slightly complicated, if the reader does not understand the underlying principle, he should not be discouraged to read it again (see also figure III).

Both sides know that there exists a bargaining range, the question is how big is it? The mediation concept used here is that each side splits a hypothetical bargaining range. First, each negotiator makes a guess about the actual bargaining range. Next, the computer asks the negotiator how much he would be willing to give to the other side for the guessed size of the bargaining range. Because it is unlikely to guess the true range, the computer uses this number as an upper limit which means that the negotiators would also accept the splitting ratio for a smaller range. This is a reasonable assumption, because in the case of a smaller range he will give less money to the other side, than initially assumed. Splitting the difference is in other words the percentage of the bargaining range the negotiator is willing to give to the other side. This information is asked from both parties independently.

Now, two conditions have to be met in order to come to an agreement. First, that the true bargaining range is smaller or equal to the assumptions of both sides and, second, both sides' willingness to split, which is described as a percentage number must add up to more than 100 percent. For example, if party A is only willing to give 35% and party B is willing to give 60% then there is still a gap of 5%, and apparently no agreement. In the case one party offers 70% and the other 55% there is an overlap, hence, they reached agreement. It seems most fair to distribute the remaining overlap evenly to both sides.

Because of the relative uncertainty of the true bargaining range and the unwillingness to offer a high percentage to the other side when the estimated range is high, the program allows the users to input a different percentage, depending on the size of the assumed gap.

The advantage of this procedure is that each negotiator knows what the upper limits for a possible agreement are. Both sides have an incentive to be generous for two reasons, the more generous the more likely it is to reach agreement and once agreement is reached the larger the overlap the more they will get back. Once the computer has calculated the final price the users can make it binding or reject it.

Fair Price Arbitration:

Whereas in the mild contract mode the reservation prices form the basis for the computer's recommendation, in this mode the fair prices create this basis (see figure IV). The reason for using this approach is that in any reasonable scenario the two fair prices are closest to each other. Hence, an agreement between these limits should be most easily reached. Asked to be the arbitrator the computer uses a special algorithm to split the difference. In the current version an attempt has been made to simulate the third parties role as a "supervisor" of fairness. The procedure uses the perceived bargaining range of each parties, this is the range between the reservation price (the one input to the machine!) and the guess of the other sides' reservation price. The suggested fair price is likely to be between these two numbers. In his fairness evaluation the third party looks at the position of the fair price. For example, if your fair price is close to the estimate of the other side you will be percieved as greedy, if it is close to your own reservation price then you are generous. The computer uses this measure to split the difference between both sides fair prices in the following way. The gap between the fair prices is divided proportionally to the anticipated profit of the two parties. So, the "greedier" party will be slightly punished by the arbitrator.

This algorithm has been chosen to somehow simulate what I think a fair arbitrator would do (?) But in addition this procedure has some other important aspects. In the beginning of the discussion I mentioned the problem of cheating. If the negotiators want to use fair price arbitration and know its algorithm algorithm then they have two counter-balancing incentives to manipulate their input data. They can either input a high reservation pricewith the risk that there is no bargaining range or they use a low fair price which would lead to a poor outcome in the arbitration. Considerations about cheating in this context have to be balanced with the part I input procedures where there was also incentive to exaggerate reservation and fair price. From my point of view it is not obvious what might be a reasonable way to cheat, at least the fair price arbitration mode complicates the situation further. A more theoretical study should determine whether the system is stable or not. The system can be described as unstable when with the uncertainty about the other side's input data, a negotiator has intentives to exaggerate his input data in either direction. In a stable system different incentives cancel each other out so that the obvious choice would be to be honest about the input data. If the current system turns out to be unstable, it would be interesting to study what conditions have to be met in order to make it stable. From my perspective, without having proven it analytically, the underlying characteristic of a prisoner's dilemma situation makes the system intrinsically stable.

The proposed system is a first attempt to let a computer act as a neutral third party in a negotiation. I have pointed at several problematic points in the system. In addition to any theoretical considerations, practicle tests have to be performed in order to gain experience about this approach. Before any real world application can be considered, the system should be extensively used in the "laboratory". The existing program would be most feasible for introductory cousres in negotiation. At MIT one small test has been done so far. Although it indicated the difficulty of providing a real incentive in a simulated negotiation, the overall response to the system was positive (Cohen, Tong 1987).

Two Party Multi-Issue Negotiation Support

ONDINE II

Structure of the program:

In the first part of the program both parties put their preference function about the negotiated issues into the computer. Then, in the actual negotiation phase they try to find a basic agreement. With the given preference functions of both players, the computer calculates the set of solutions which have a higher utility (Pareto efficient solutions) than the outcome reached so far (intermediate agreement). Then both parties have to negotiate which point out of the set of Pareto efficient solutions shall be the final agreement (post-settlement settlement).

Part I Input of utility function:

In this program utility is represented in a discrete rather than continuous form. Before the actual negotiation, the parties have to agree on the number of issues they would like to negotiate about and for each issue a set of possible outcomes, the so-called solution space. For example, if one of the negotiated issues is the number of aircraft carriers, possible solutions in an agreement are zero, one, two, etc. For each outcome, in our example it is the number of aircraft carriers the parties have to determine the utility associated with it (cost of the carrier, economies of scale etc.). This set of numbers can be called utility space. The procedure has to be applied for all points in the solution space.

The case of the aircraft carrier also illustrates the usefulness of a discrete utility function, it would make little sense to negotiate about two and a half carriers. In most negotiations the issues discussed are given in a discrete form. in cases of continuous utility functions, these can be simulated accurately as step functions.

The current program requires that both parties can relatively precicely quantify their utility function. This premise is usually given when all issues can be measured in one unit e.g. money. But in some cases, the assessment of utility can cause considerable problems. Another limitation of this representation of the utility function is that all issues must be independent of each other, which is also sometimes not realistic but it allows to add utilities and to calculate and compare utilities in a relatively simple way, so that it can be understood by people who are not experts in fields like decision analysis.

Nevertheless, inspite of these drawbacks, this approach has the advantage of simplicity. In face of the degree of uncertainty in a negotiation, demanding a highly accurate utility function would be focusing on the wrong end of the problem.

The main objective of the program is to test the potential of the computer to act as an independent third party. To my knowledge this is the first attempt; with its operational features the program has been designed to assist negotiation with about five to ten issues (although the program can theoretically handle a much larger number of issues) which all can be reasonably represented in economic terms i.e. that they can be measured in one unit.

Part II Negotiation:

In this part the computer still plays a relatively passive role. Its main function is to calculate for each party the resulting utility of any particular offer. A negotiation round starts with each side putting together an offer that contains a set of all issues. This package is sent to the other side. Both offers are displayed to the other side simultaneously. The computer also recalculates the other side's offer in terms of the own utility measure. Of course, this number remains private. The program assumes that an agreement is possible and that after several iterations the parties reach an intermediate agreement.

Part III Improving the agreement:

At this point the computer compares the intermediate agreement (end of part II) with the solutions on the pareto frontier, which have been internally calculated at an earlier point. For this calculation I use an optimization routine (see Blass and Raiffa 1987), which uses the utility functions of both sides and iteratively arrives at the set of Pareto efficient solutions. If the the existing intermediate agreement is not Pareto optimal, the computer notifies the parties and asks them whether they would like to try to negotiate an improvement, using the computer's knowledge about the improved outcomes.

As laboratory tests have shown, even in relatively simple multi-issue negotiations the parties usually do not come close to the efficiency frontier.

The problem for the parties is to agree on a point on the frontier which satisfies both. The set of Pareto solutions ranges from no improvement for one side and a large improvement for the other to the opposite (see figure V). Discussions about fairness in economic terms (Raiffa 1982) show that there are many answers to what can be rationally perceived as fair, but which all lead to quite different outcomes.

In this new context, where the computer represents a neutral party, it seems to me most important that the negotiators feel independent of any unneccessary built-in fairness procedures of the computer. Thus, I simply ask the parties whether they would like to see the set of pareto efficient outcomes. If both agree, then they are displayed to each side and they have to negotiate which point to select as the final agreement.

There are several problems with this approach; for example, both know their rivals "reservation utility" and it is not any longer possible to use other bargaining chips (at least not in the initial framework of issues). In other words, the multi-issue negotiation has been transformed into a single-issue one, where "utility" is the new issue. Nevertheless, I think that at first, experience with this system is necessary before more complex and more satisfactory strategies should be developed. At this point of the development of computer assisted negotiation, it is simply too hard to predict, in what way negotiators would like the computer to act. Following this section, I will suggest strategies for more complex Computer Third Party systems.

Part IV New Approaches:

As improvements for the current version of the system I focus on two areas, a) the negotiation phase and b) the post-settlement settlement phase, both which are quite different in character.

a) Improvements for the Negotiation Phase:

At least in theory a multi-issue case can be transformed into a single issue case by introducing utility as a "meta-issue". I say "in theory" because it assumes that the negotiators are consistent with their utility assessment during the course of the negotiation, which is rarely the case in reality. Nevertheless, there might be situations where there is little deviance from the theortical assumptions. Then the computer could assist the negotiation in a way similar to the single issue case (see ONDINE 1) with the following functions:

- Find out about bargaining range by asking for "reservation utility" (see figure VI); there is a positive bargaining range when the intersection of both parties' reservation utilities (which is a line in the utility plane) is under the pareto curve.
- For each set of offers the computer could find out whether their intersection is under the pareto frontier or not. If it is

the case, then the machine could notify the parties that it is possible to reach agreement at this level of utility although with a different composition of the issues. If the parties wish, the Computer Third Party could then suggest a solution. Note,

that such a proposal is not necessarily a Pareto efficient one!

- The computer could also tell the parties which pair of issues promises a high trade-off that has not been exploited yet. But this procedure bears the problem that it reveales, to some extent at least, the parties' utility functions, which might not be desirable.
- Other mediation and arbitration procedures (similar to ONDINE I) where the parties let the computer split a gap in a certain way.

b) Improvements for Post-Settlement Settlement:

One important practical consideration is the number of possible solutions; in a continuous system this number is infinite. In cases where solutions are discrete, the Pareto frontier is sometimes very unevenly covered with possible outcomes. Often this problem can be overcome by randomizing the discrete points, i.e. creating a continuous solution space, but the aircraft carrier shows the limits of this procedure.

Here I suggest only one new procedure for post-settlement settlement. In order to avoid displaying all points to the parties, the computer asks them in what way would they agree to split the improvment. Figure VII illustrates this procedure, essentially the parties negotiate the angle of the line that intersects with the

Pareto frontier determining the final outcome. The problem arises when in a descrete case there is no solution that satisfies this conditions and randomization is not possible. To me the feature that makes this procedure attractive is that the parties are not forced by the computer to any fairness principle, rather they determine it themselves.

Conclusions

The two programs, ONDINE I and II, which I introduce in this paper are only a first step in the development of using computers as independent third parties. They have been mainly designed for academic purposes, to study and teach the principles of negotiation and to investigate the potential of the computer to act as a neutral advisor helping the parties to come to better agreements. The programs as they are right now would probably be an ideal supplement for classes teaching negotiation. Even without any changes they could be used in the standard negotiation case studies. Experience in a laboratory environment would give important information about the usefulness but also about the limitations of this approach. A basic study which should be done is to compare the outcomes of one negotiation case with and without the use of the computer. Research of this type would be necessary to develop new strategies for the computer to support negotiations. It would also indicate what type of negotiations are most feasible for the computer as a third party.

Possibly these two programs can even be used in a real negotiation without major changes. In this case it would be useful to adapt them to more portable personal-computer systems. The current versions are installed on Digital Equipment VAX II workstations using the X-10 window system as an communication environment.

I hope that this project will stimulate the ongoing discussion about the use of computers in negotiation and decision making and give rise to fresh ideas.

References

A. Blass, H. Raiffa. 1986. "Computer Program for investigating the efficient solutions of two-party, multiple-issue negotiations". Harvard University 1986.

J. Cohen, R. Tong. 1987. "Usability and effectiveness of computer mediation in single issue negotiations".
MIT, Sloan School of Management, 1987.

J. D. Kettelle et al. 1986. "Systems analysis and mathematical modeling in arms control".
OR Spectrum 1986, 8:129-141

-- 1986,2. Personal computer based program to demonstrate the concept of the computer as a third party. (unpublished)

J. D. Nyhart, Ch. Goeltner. 1987. "Computer models as support for complex negotiations."
31st annual meeting of the international society for general systems research, Hungary June 1987.

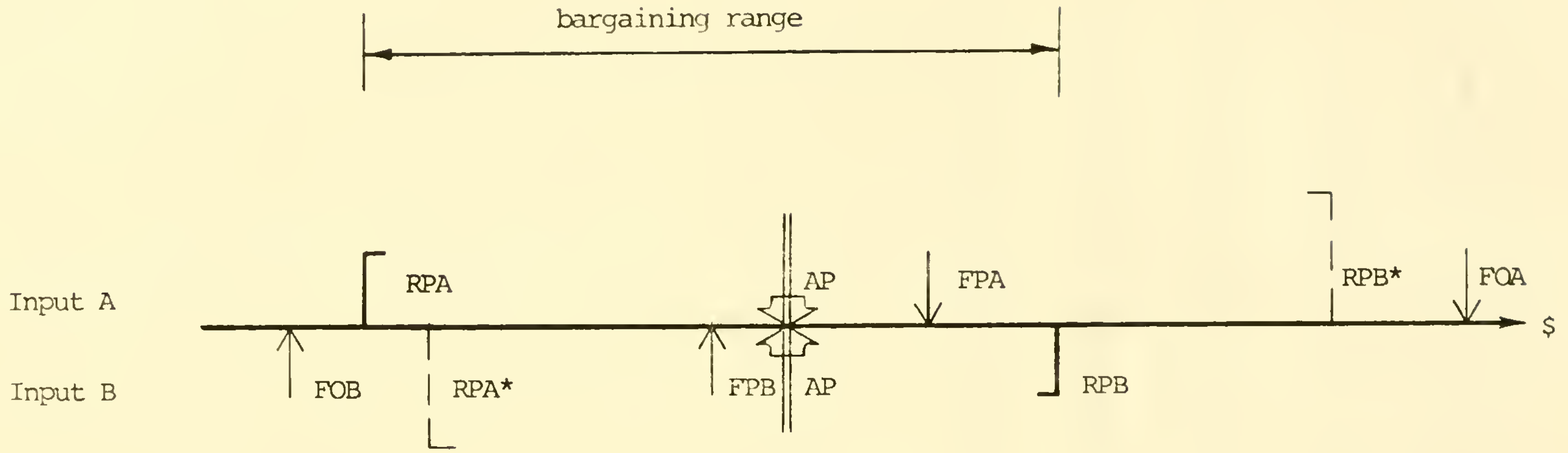
H. Raiffa. 1982. "The art and science of negotiation." See chapter 16, arbitration of disputes.
Belknap Press, Harvard, 1982.

J. Rohrbaugh. 1980. "Measuring the relative importance of utalitarian and egalitarian values: A study of individual differences about distribution."
Journal of applied psychology, Vol. 65 #1 Feb. 1980.

J. K. Sebenius. 1981. "The computer as mediator: The law of the sea and beyond."
Journal of policy analysis and management. Vol.1 No.1,77-95, 1981

L. Susskind. 1986. "Diplomat", Harvard negotiation newsletter, 1-1986

Figure I basic parameters of a two party, single-issue negotiation



A: Seller
B: Buyer

RPA/RPB	Reservation Price Seller/Buyer
RPA*/RPB*	Buyer's/Seller's estimate of Seller's/Buyer's Reservation Price
FPA/FPB	Fair Price Seller/Buyer
FOA/FOB	First Offer Seller/Buyer

Figure II Determination of Fair Price

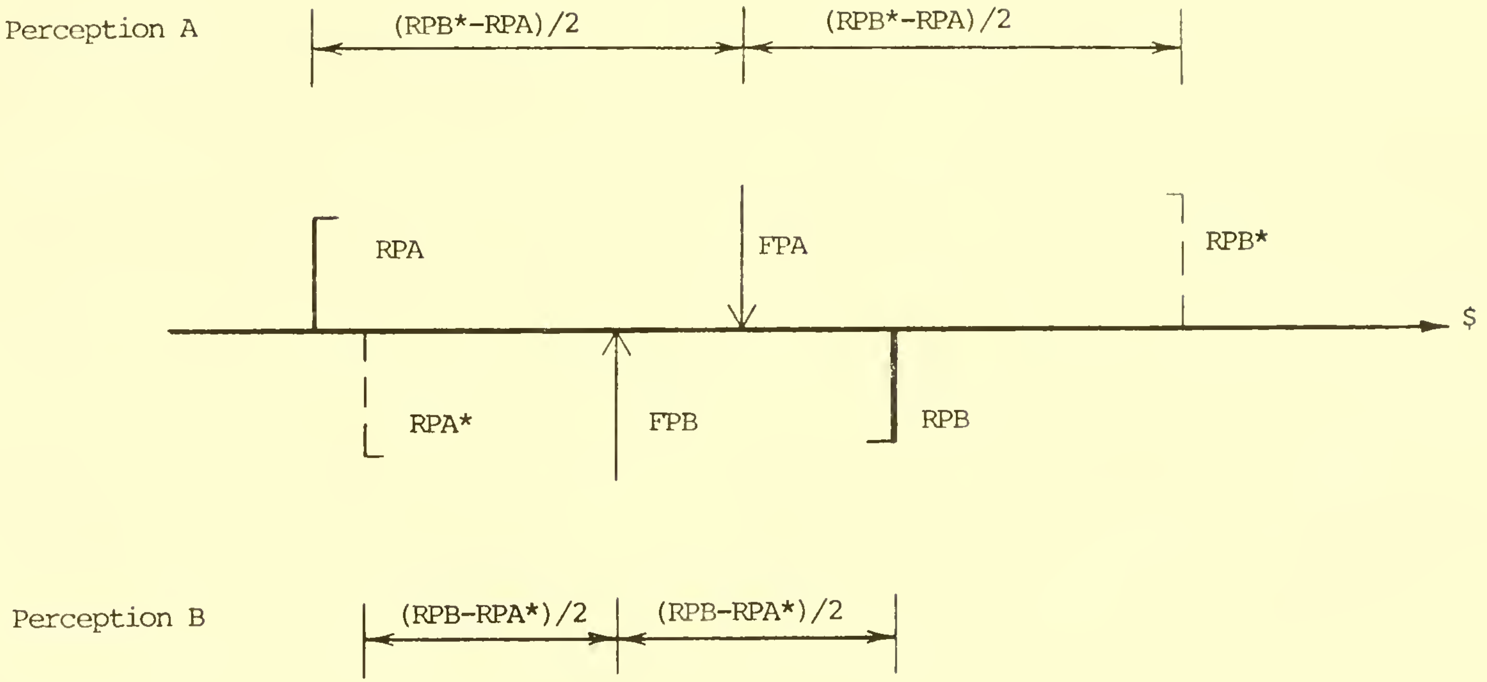
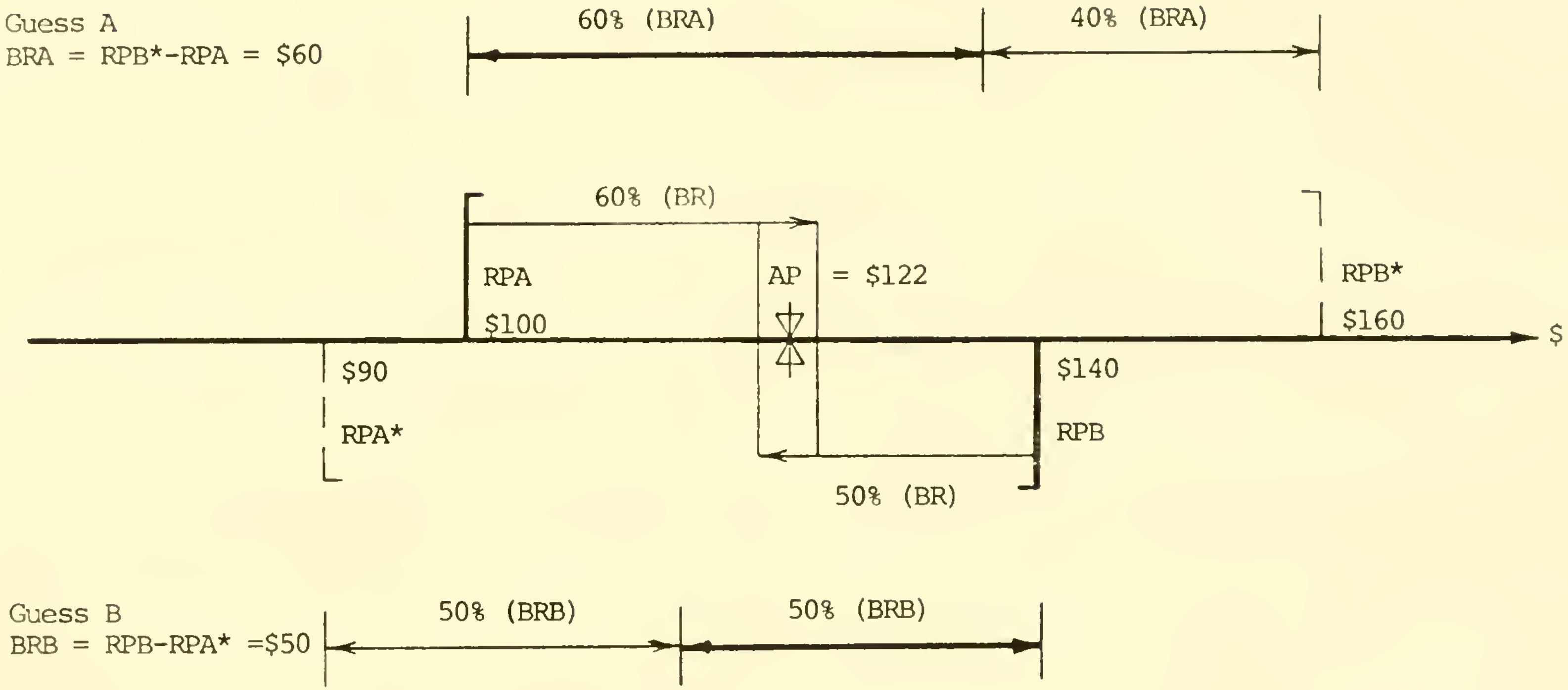


Figure III Mild Contract



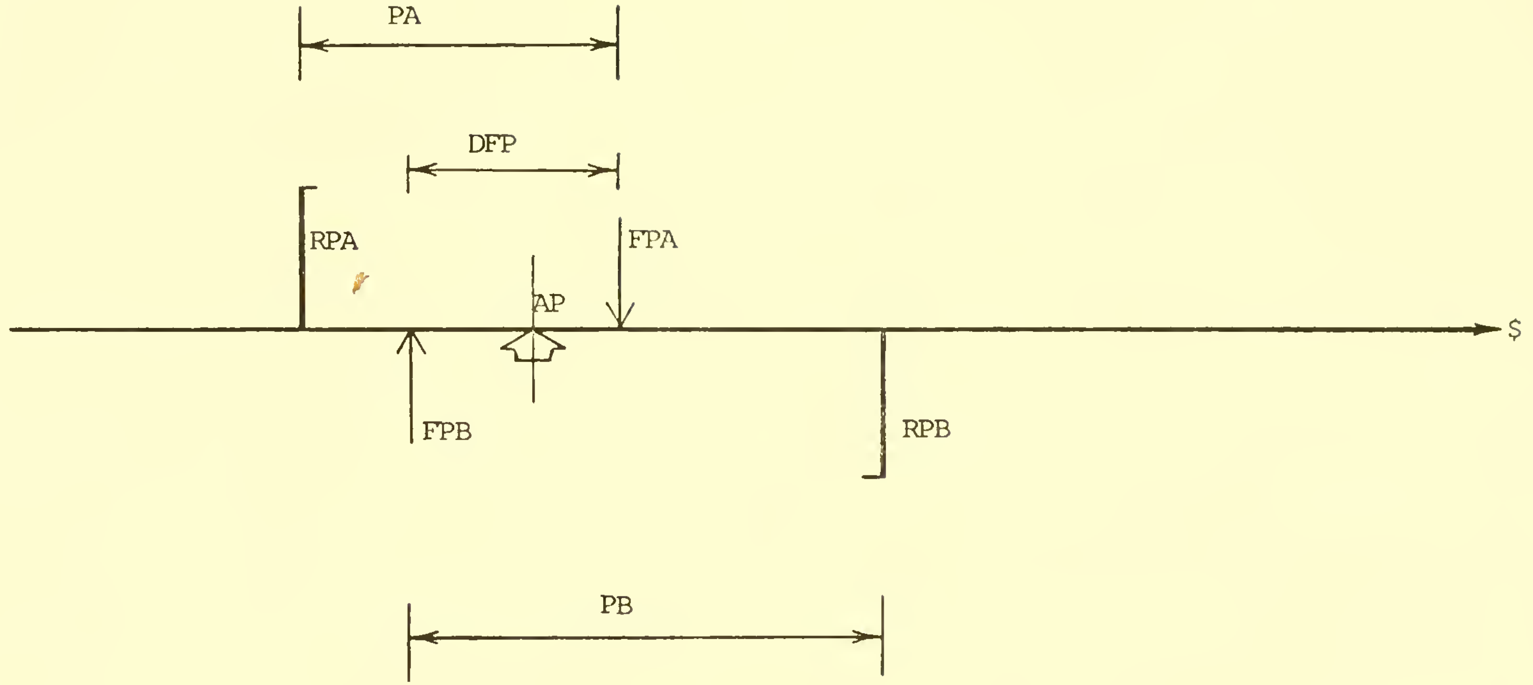
BRA	A's guess of Bargaining Range
BRB	B's guess of Bargaining Range
BR	true bargaining Range

For the given example:
BR = $40
BRA = $60 percentage willing to give to B: 60%
BRB = $50 percentage willing to give to A: 50%

Total : 110% overlap : 10%

$$\text{Agreement Price} = AP = RPA+50\%(BR)+1/2\text{overlap}(BR)$$
$$= \$100+50\%(\$40)+1/2[10\%(\$40)]$$
$$= \$122$$

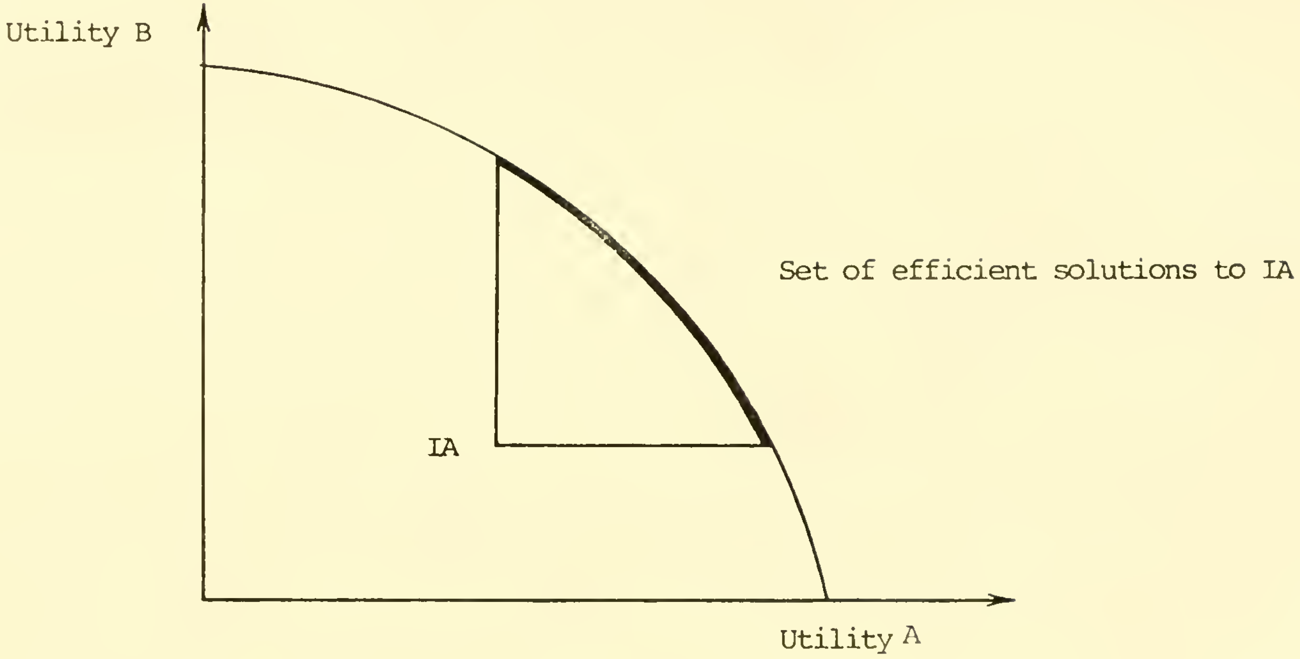
Figure IV Fair Price Arbitration



BR = Bargaining Range = RPB - RPA
DFP = Difference between Fair Prices = FPA - FPB
PA = Profit A = FA - RPA   relative Profit = rPA = PA/BR
PB = Profit B = RPB - FB   relative Profit = rPB = PB/BR

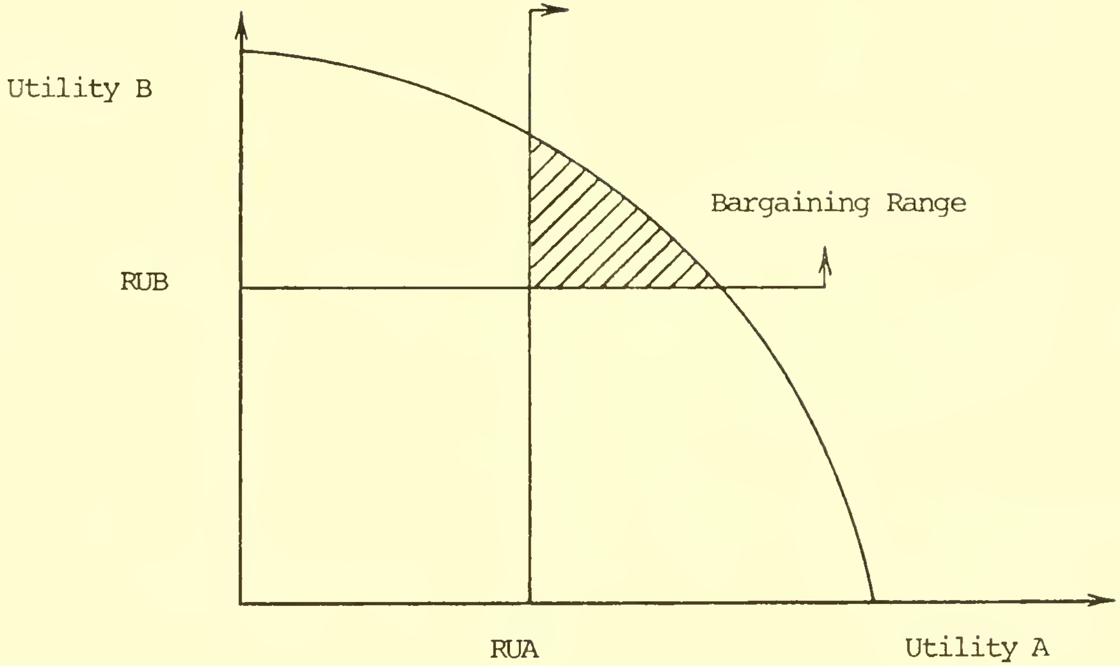
Fair Price Arbitration: AP = FPB+DFP/2+[(rPB-rPA)DFP]

Figure V Pareto Efficiency Frontier in Two Party Multi-Issue Negotiation



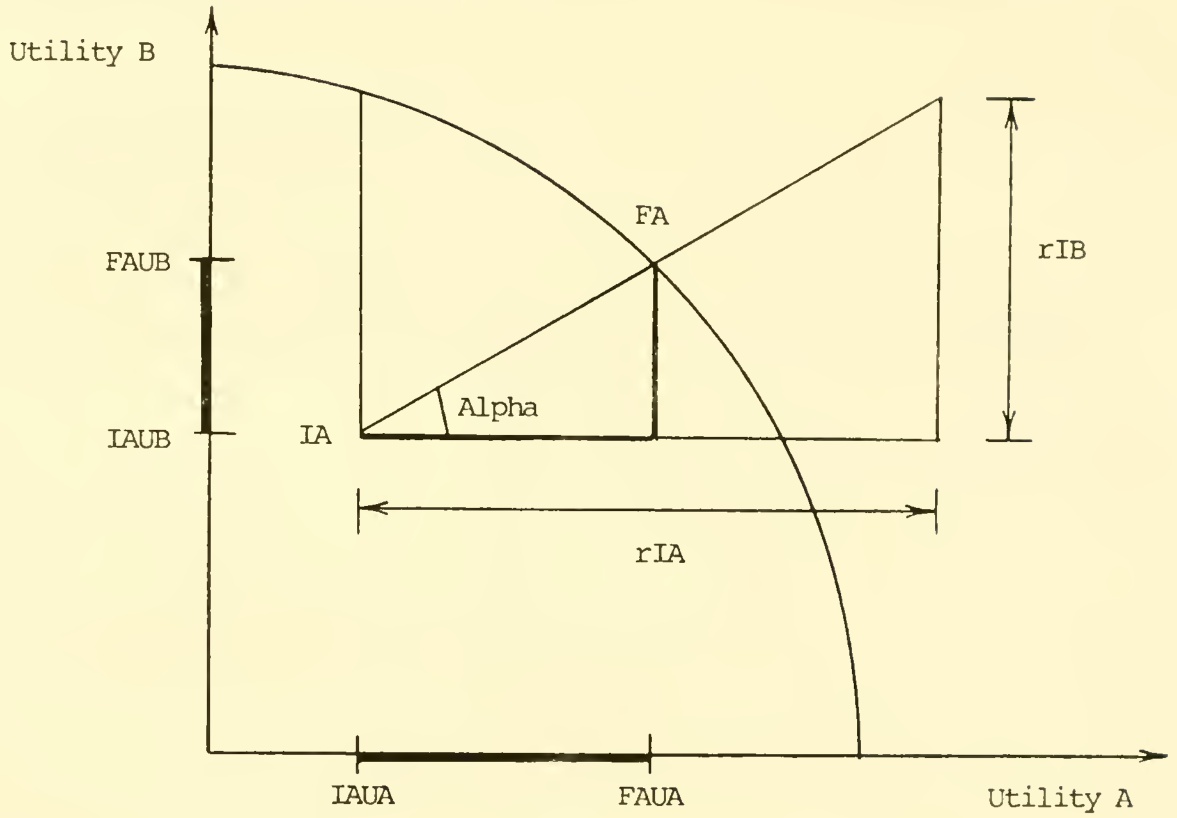
IA   Intermediate Agreement

Figure VI Bargaining Range for Two Party Multi-Issue Negotiation



RUA	Reservation Utility A
RUB	Reservation Utility B

Figure VII Post-Settlement Settlement Strategy



Intermediate Agreement (IA): Utility for A = IAUA; Utility for B = IAUB

Final Agreement (FA): Utility for A = FAUA; Utility for B = FAUB

Improvement of Intermediate Agreement: relative Improvement for A = rIA
relative Improvement for B = rIB
absolute Improvement for A = FAUA - IAUA
absolute Improvement for B = FAUB - IAUB

tangens (alpha) = rIB/rIA

4534 093

Date Due

DE 06 '88

P 23 1991

JAN 17 [illegible]

[illegible]

MAY 16 198[illegible]

Lib-26-67

MIT LIBRARIES

3 9080 005 130 965